NORTH-HOLLAND

Fast Transforms of Toeplitz Matrices

Martin Ohsmann
Philips Research Laboratory
P. O. Box 1980
D-52021 Aachen, Germany

Submitted by Ludwig Elsner

ABSTRACT

We consider the problem of computing elements of the product $\hat{A} = TAS^T$, where A is an $N \times N$ Toeplitz matrix and T and S are matrices denoting Fourier-transform or cosine-transform matrices. We prove that it is possible to compute p elements of $\hat{A}$ in time $O(p + N \log N)$ with only $O(N)$ auxiliary storage. [Classical application of FFT techniques need $O(p + N^2 \log N)$ time and $O(N^2)$ storage.] The algorithm is not restricted to square matrices, but can handle circulant or Hankel matrices also. The algorithm is especially useful if only some of the N^2 elements of $\hat{A}$ have to be computed. Even if all elements have to be computed, the algorithm is faster than traditional methods. Some applications are discussed.

1. INTRODUCTION

In digital signal processing a number of discrete transforms are used [3, 5, 7]. These transforms map a vector $\vec{x} = (x_0, x_1, \ldots, x_{K-1})^T$ onto the transformed vector $\vec{X} = (X_0, X_1, \ldots, X_{L-1})^T$. The best-known transform is the discrete Fourier transform (DFT) of size N given by

$$X_l = \sum_k x_k \omega^{kl} \tag{1}$$

with $K = L = N$ and $k = 0, \ldots, K-1$, $l = 0, \ldots, L-1$, and $\omega = e^{-2\pi i/N}$. We often speak of $\vec{x}$ as the time-domain representation, and of $\vec{X}$ as the frequency-domain representation.

0024-3795/95/$9.50
SSDI 0024-3795(94)00043-D

Many problems in this field lead to linear systems or eigenvalue problems. These problems can also be discussed in the time and in the frequency domain. To change the domain, the matrices involved have to be transformed correspondingly. Often the matrices have Toeplitz structure $[A = (a_{lk}) = (a_{k-l})]$ in the time domain, and in the following we show how to transform them very fast. It is a well-known fact that circulant matrices can be diagonalized by DFT matrices. This observation leads to the concept used in this paper. We discuss not only a generalized DFT but also cosinelike transforms. The results presented in this paper are generalizations of the result in [4] where only a special cosine transform of a square Toeplitz matrix was treated.

2. THE CHIRP z TRANSFORM

The so-called *Chirp z transform* (CZT) forms the basis of our algorithm [1, 6]. The CZT $\vec{X} = (X_0, X_1, \dots, X_{L-1})^T$ of a (complex) vector $\vec{x} = (x_0, x_1, \dots, x_{K-1})^T$ is defined by

$$X_l = \sum_k x_k \zeta^{kl}, \tag{2}$$

where ζ is an arbitrary complex number. We call (2) a CZT of length $L + K$ and use the following theorem [6]:

THEOREM 1. *The L quantities X_l can be computed using $O((L + K)\log(L + K))$ time from the K quantities x_k. The auxiliary storage needed is $O(L + K)$.*

The task of performing this CZT can be reduced to the computation of two DFTs of size N with $N > K + L$. Using the FFT technique for this task yields Thoerem 1, as shown in [6]. The same theorem holds for a slightly generalized CZT defined by

$$X_l = \sum_k x_k \zeta^{(k+a)(l+b)+c}, \tag{3}$$

where ζ, a, b, and c are arbitrary complex numbers. This is due to the fact that (3) may be interpreted as pre- and postmultiplication of the vectors involved:

$$X_l = \zeta^{a(l+b)+c} \sum_k \zeta^{kl}\left(x_k \zeta^{kb}\right). \tag{4}$$

This generalized CZT will form the base algorithm used throughout this paper. The classical DFT is a special case of a CZT. The constants hidden in the O symbols are fairly small, so that the algorithm is applicable in practice even with only moderately large $K + L$. (The CZT approach is also useful in performing a fast order-N DFT if N is neither a power of 2 nor highly composite and classical FFT approaches fail.)

3. CHIRP z TRANSFORM OF TOEPLITZ MATRICES

Let A be an $L \times K$ Toeplitz matrix

$$A = \begin{pmatrix} a_0 & a_1 & \cdots & a_{K-1} \\ a_{-1} & a_0 & \cdots & a_{K-2} \\ \vdots & \vdots & \ddots & \vdots \\ a_{1-L} & a_{2-L} & \cdots & a_{K-L} \end{pmatrix}.$$

We now consider the transformed matrix $\hat{A}$ which results from transforming A using the two CZT matrices T and S. We define $T = (t_{m,l})$ and $S = (s_{n,k})$ by

$$t_{m,l} = \tau_m \xi^{(m+\beta)(l+\alpha)+\gamma}, \qquad s_{n,k} = \sigma_n \zeta^{(n+b)(k+a)+c}, \tag{5}$$

where the constants involved are again arbitrary and the constants τ_m and σ_n may be necessary for normalization purposes. The transformed matrix $\hat{A}$ is now defined by

$$\hat{A} = TAS^T = (\hat{a}_{m,n}). \tag{6}$$

Throughout this paper the index ranges are as follows if not noted explicitly:

$$k = 0, \ldots, K-1, \qquad l = 0, \ldots, L-1,$$
$$m = 0, \ldots, M-1, \qquad n = 0, \ldots, N-1, \tag{7}$$

A comma is inserted between double subscripts to clarify reading, since sometimes arithmetic expressions appear as subscripts. So S is an $N \times K$ matrix, A an $L \times K$ matrix, and so on. (In practice the case $K = L = M = N$

with $S^T = T^{-1}$ is of special interest, since it preserves the eigenvalues.) Our main task is to prove:

THEOREM 2. *The computation of* p *quantities* $\hat{a}_{m,n}$ *can be accomplished using* $O(p + q \log q)$ *time with* $q = \max(K, L, M, N)$. *The auxiliary storage needed is* $O(q)$.

We proceed starting with

$$\hat{a}_{m,n} = \tau_m \sigma_n \sum_k \sum_l t_{m,l} a_{l,k} s_{n,k}. \tag{8}$$

The constants τ_m and σ_n introduce no additional complexity and are therefore omitted in the following. We introduce auxiliary quantities $x_{m,k}$ by

$$x_{m,k} = \sum_l \xi^{(m+\beta)(l+\alpha)+\gamma} a_{l,k} \tag{9}$$

and get

$$\hat{a}_{m,n} = \sum_k x_{m,k} \zeta^{(n+b)(k+a)+c}. \tag{10}$$

As a next step we define the finite z transform of the quantities $x_{m,k}$ by

$$X_m(z) = \sum_k x_{m,k} z^k \tag{11}$$

and get the following representation for the elements of $\hat{A}$:

$$\hat{a}_{m,n} = \zeta^{(n+b)a+c} \sum_k x_{m,k} \zeta^{(n+b)k} = \zeta^{(n+b)a+c} X_m\left(\zeta^{(n+b)}\right). \tag{12}$$

Equation (12) shows that $\hat{A}$ may be computed using N CZTs, resulting in an algorithm using $O(q^2 \log q)$ time and q^2 additional storage. This is the classical approach in decomposing two-dimensional transforms.

Up to this point we have made no use of the Toeplitz structure of A. Now we will exploit the structure of A to derive an algorithm which is more efficient. Since A has Toeplitz structure, we have $a_{l,k} = a_{k-l}$. Because each column of A is a circular shift of the preceding column plus two updates at the boundaries, we can derive a recurrence relation:

$$
\begin{aligned}
x_{m,k+1} &= \sum_{l=0}^{L-1} \xi^{(m+\beta)(l+\alpha)+\gamma} a_{k+1-l} \\
&= \sum_{j=-1}^{L-2} \xi^{(m+\beta)(j+1+\alpha)+\gamma} a_{k-j} \\
&= \xi^{(m+\beta)} \sum_{j=-1}^{L-2} \xi^{(m+\beta)(j+\alpha)+\gamma} a_{k-j} \\
&= \xi^{(m+\beta)} \Bigg(\sum_{j=0}^{L-1} \xi^{(m+\beta)(j+\alpha)+\gamma} a_{k-j} \\
&\qquad + \xi^{(m+\beta)(-1+\alpha)+\gamma} a_{k+1} - \xi^{(m+\beta)(L-1+\alpha)+\gamma} a_{k-L+1} \Bigg). \qquad (13)
\end{aligned}
$$

So the auxiliary quantities $x_{m,k}$ satisfy the recurrence relation

$$
x_{m,k+1} = \xi^{(m+\beta)} x_{m,k} + \xi^{(m+\beta)\alpha+\gamma} \left(a_{k+1} - \xi^{L(m+\beta)} a_{k-L+1} \right). \qquad (14)
$$

This recurrence is valid for $k = 0, \dots, K-2$, $m = 0, \dots, M-1$. We extend its range to $k = K-1$. This may be considered as augmenting A by an additional column containing the additional element a_K. We may assign an arbitrary value to this quantity; this defines $x_{m,K}$. (It is numerically preferable to choose $a_K = 0$.) The recurrence relation has a separated form in the sense that the indices k and m appear only in separate factors in products. This separation and the fact that the recurrence relation is linear with constant coefficients lead to the fast algorithm we want to derive.

Now consider the z transform of the relation (14). Transformation of the left side yields

$$
\begin{aligned}
\sum_{k=0}^{K-1} x_{m,k+1} z^k &= \sum_{k=-1}^{K-2} x_{m,k+1} z^k - x_{m,0} z^{-1} + x_{m,K} z^{K-1} \\
&= \sum_{k=0}^{K-1} x_{m,k} z^{k-1} - x_{m,0} z^{-1} + x_{m,K} z^{K-1} \\
&= z^{-1} X_m(z) - x_{m,0} z^{-1} + x_{m,K} z^{K-1}. \qquad (15)
\end{aligned}
$$

Using

$$U(z) = \sum_k a_{k+1} z^k \quad \text{and} \quad V(z) = \sum_k a_{k-L+1} z^k, \tag{16}$$

we transform the right side of (14) and get

$$\sum_{k=0}^{K-1} x_{m,k+1} z^k = \xi^{(m+\beta)} X_m(z) + \xi^{(m+\beta)\alpha+\gamma} U(z) - \zeta^{(m+\beta)(L+\alpha)\gamma} V(z). \tag{17}$$

Using these results, we get

$$X_m(z) = \frac{1}{z^{-1} - \xi^{(m+\beta)}} \Big[x_{m,0} z^{-1} - x_{m,K} z^{K-1} + \xi^{(m+\beta)\alpha+\gamma} U(z) - \xi^{(m+\beta)(L+\alpha)+\gamma} V(z) \Big], \tag{18}$$

which is again a separated form, this time with respect to the variables m and n.

To compute $\hat{a}_{m,n}$ we have to evaluate this function at the N values $z_n = \zeta^{(n+b)}$. This can be accomplished by using the CZT. First we compute $x_{m,0}$ and $x_{m,K}$ by CZTs of length $L + M$ using Equation (9). Then we evaluate $U(z)$ and $V(z)$ according to (16) at the points z_n, which can be done with two CZTs of length $K + N$. After these preliminary steps all values of $\hat{a}_{m,n}$ can be computed in constant time using (18). This proves Theorem 2 in the case where the denominator of (18) is not zero. To handle this singular case we note that $X_m(z)$ is a continuous fucntion at $z_0 = \xi^{-(m+\beta)}$ and we may use l'Hospital's rule to determine its value:

$$X_m(z_0) = \frac{(d/dz)[x_{m,0} - x_{m,K} z^K + \xi^{(m+\beta)\alpha+\gamma} z U(z) - \xi^{(m+\beta)(L+\alpha)+\gamma} z V(z)]}{(d/dz)\{1 - z\xi^{(m+\beta)}\}} \Bigg|_{z=z_0}$$

$$= \frac{-K x_{m,K} z^{K-1} + \xi^{(m+\beta)\alpha+\gamma}(zU(z))' - \xi^{(m+\beta)(L+\alpha)+\gamma}(zV(z))'}{-\xi^{(m+\beta)}} \Bigg|_{z=z_0}. \tag{19}$$

This results in the additional formula

$$X_m(z_0) = Kx_{m,K} z_0^K - \xi^{(m+\beta)(\alpha-1)+\gamma}[zU(z)]'|_{z_0} + \xi^{(m+\beta)(L+\alpha-1)+\gamma}[zV(z)]'|_{z_0} \tag{20}$$

with

$$[zU(z)]'|_{z_0} = \sum_k (k+1) z_0^k a_{k+1} = \sum_k \xi^{-(m+\beta)k}(k+1)a_{k+1} \tag{21}$$

and

$$[zV(z)]'|_{z_0} = \sum_k (k+1) z_0^k a_{k-L+1} = \sum_k \xi^{-(m+\beta)k}(k+1)a_{k-L+1}. \tag{22}$$

The evaluation has to be performed only for $m = 0, \dots, M-1$; this again can be done using a length $(K+M)$ CZT. So the proof of Theorem is complete, and we can state the algorithm as follows:

ALGORITHM. (CZT of a Toeplitz matrix).

A. Setup:
1. Compute x_{m0} for $m = 0, \dots, M-1$ as the CZT of $(a_0, \dots, a_{1-L})$.
2. Compute x_{mK} for $m = 0, \dots, M-1$ as the CZT of $(a_K, \dots, a_{K+1-L})$.
3. Compute $U(z_n)$ with $z_n = \zeta^{(n+b)}$ for $n = 0, \dots, N-1$ as the CZT of $(a_1, \dots, a_K)$.
4. Compute $V(z_n)$ with $z_n = \zeta^{(n+b)}$ for $n = 0, \dots, N-1$ as the CZT of $(a_{1-L}, \dots, a_{K-L})$.

If $\zeta^{-(n+b)} = \xi^{(m+b)}$ for some n, m then:

5. Compute $[zU(z)]'_{z_0}$ with

$$z_0 = \xi^{-(m+b)} \qquad \text{for} \quad m = 0, \dots, M-1$$

as the CZT of $(1a_1, 2a_2, \dots, Ka_K)$.

6. Compute $[zV(z)]'_{z_0}$ with

$$z_0 = \xi^{-(m+b)} \qquad \text{for} \quad m = 0, \dots, M-1$$

as the CZT of $(1a_{1-L}, 2a_{2-L}, \dots, Ka_{K-L})$.

B. Computation of p elements $\hat{a}_{m,n}$:
   7. If $\zeta^{-(n+b)} \neq \xi^{(m+b)}$ use (18) else use (20) to compute $\hat{a}_{m,n}$ for the given m, n.

Complexity:

Steps 1 and 2: 2 CZTs of length $M + L$.
Steps 3 and 4: 2 CZTs of length $N + K$.
Steps 5 and 6: 2 CZTs of length $M + K$.
Step 7: p Const operations.

Total: $O(p + q \log q)$ with $q = \max(K, L, M, N)$.

For small values of K, L, M, and N the direct implementation of the CZTs may be faster than the FFT approach. If the CZTs are implemented directly, the time needed is $O(p + q^2)$.

4. COSINE TRANSFORMS OF TOEPLITZ MATRICES

Many real transformation used in signal processing can be represented in the form

$$t_{m,l} = \tau_m \cos[(m + \beta)(l + \alpha)\varphi + \gamma] \tag{23}$$

with real constants τ_m, β, α, φ, and γ. This includes the various discrete sine and cosine transforms and the Hartley transform [5]. In the following we consider a real Toeplitz matrix A. (For complex matrices the transforms of the real and imaginary parts may be performed separately.)

We consider the following generalized transforms:

$$t_{m,l} = \tau_m \Re\{\xi^{(m+\beta)(l+\alpha)+\gamma}\}, \qquad s_{n,k} = \sigma_n \Re\{\zeta^{(n+b)(k+a)+c}\}. \tag{24}$$

(Here $\Re\{\cdot\}$ denotes the real part of a complex quantity.) If the auxiliary quantities $x_{m,k}$ and $X_m(z)$ are defined as above, this yields

$$\hat{a}_{m,n} = \sum_k \Re\{x_{m,k}\}\Re\{\zeta^{(n+b)(k+a)+c}\} = \Re\left\{\sum_k \tfrac{1}{2}(x_{m,k} + \bar{x}_{m,k})\zeta^{(n+b)(j+a)+c}\right\} \tag{25}$$

The constant factors τ_m and σ_n are omitted to simplify reading. These constants present no additional computational complexity. Using $X_m(z)$ we get the representation

$$\hat{a}_{m,n} = \tfrac{1}{2}\Re\left\{\zeta^{(n+b)a+c}\left[X_m\left(\zeta^{(n+b)}\right) + \overline{X_m\left(\bar{\zeta}^{(n+b)}\right)}\right]\right\}. \tag{26}$$

This formula is only slightly more complex than (18). The additional cost is due to the fact that $X_m(z)$ has to be evaluated additionally at the values $z'_n = \bar{\zeta}^{(n+b)}$. This can be accomplished with two additional CZTs for the evaluation of $V(z)$ and $U(z)$ at these points. This proves Theorem 2 also for the case of real transforms represented by (24).

5. EXTENSIONS

The derivations presented can be carried over to the case of Hankel matrices. The case of a CZT of a matrix with elements $a_{k,l} = \rho^k b_{l-k}\delta^l$ can also be handled, since the powers of ρ and δ can be absorbed in the transforms. The case of circulant matrices is included, since no symmetry restrictions are imposed on the Toeplitz matrices. Algorithms for the fast transform of matrices which are generated by other updates or other shift structures can be obtained by the same technique. This includes the case of products of Toeplitz matrices and their inverses.

6. APPLICATIONS

If the transformation matrices satisfy $T^{-1} = S^T$, the eigenvalues are invariant under the transformation. It can be observed that the transformation tends to concentrate the large elements of $\hat{A}$ around a part of the diagonal. This property [2] can be used for some applications described in [4].

Another application is the least-squares problem treated in [8]. The problem is to minimize

$$\|M\vec{x} - \vec{y}\|$$

where $\vec{y}$ is a given data vector, the entries of M are measured quantities, and the N entries of $\vec{x}$ can be interpreted as samples of a function to be estimated. The problem in this case [8] has the property that $A = M^TM$ is a Toeplitz matrix. If the problem is solved directly (using a fast Toeplitz solution method), the result is very poor, as can be seen from Fig. 1. This

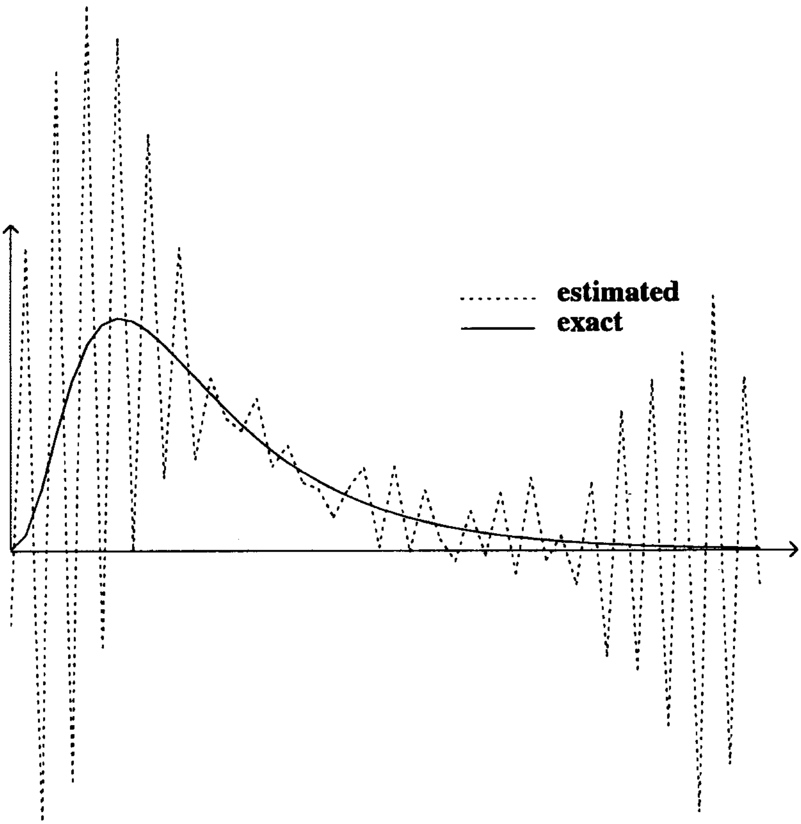

FIG. 1. Estimation based on an LS method using a classic Toeplitz solver.

behavior is due to the fact that the problem is ill posed. Another approach is to restrict the minimization space to the column space (of dimension $\tilde{N} < N$) of a matrix T. The resulting normal equations are now

$$T^T M^T M T \vec{z} = T^T A T \vec{z} = T^T M^T \vec{y} \qquad \text{with} \quad \vec{x} = T\vec{z}.$$

If we choose $T = (t_{m,l})$ with $t_{m,l} = e^{-\alpha m l}$, the matrix T represents CZT and the matrix A can be transformed very fast as shown above. The result of this approach is shown in Fig. 2, which demonstrates the applicability of this method. In this example $\tilde{N} \approx N/5$ holds, so that the size of the system to be solved reduces considerably. (Unfortunately, the transformed system has no longer Toeplitz structure, so no fast solver can be used.)

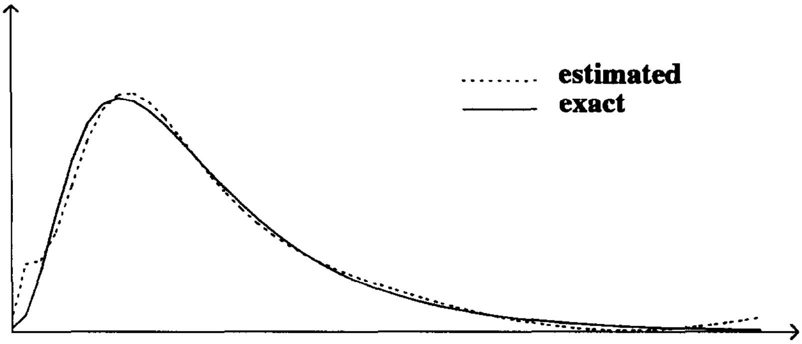

FIG. 2. Estimation based on a fast transformation with subspace restriction.

7. CONCLUSIONS

An algorithm for fast transforms of Toeplitz, Hankel, and circulant matrices has been derived. The method is not only fast but also storage efficient. It facilitates the computation of selected elements of the transformed matrix without storing any matrix. So it is possible to solve problems in the transformed domain also if the dimension of the matrices is large and a complete matrix storage impossible. The ideas used in the derivation may also be used for other transformations and other matrix types. The algorithm derived is of practical interest in the field of signal processing, as examples show.

REFERENCES

1 A. V. Oppenheim and R. W. Schafer, *Digital Signal Processing*, Prentice-Hall, 1975.
2 R. M. Gray, On the Asymptotic eigenvalue distribution of Toeplitz matrices, *IEEE Trans. Inform. Theory* IT-18, No. 6 (Nov. 1972).
3 D. F. Elliot and K. R. Rao, *Fast Transforms*, Academic, 1982.
4 M. Ohsmann, Fast cosine transform of Toeplitz matrices, algorithm and applications, *IEEE Trans. Signal Process.* 41(10):3057 (Oct. 1993).
5 D. G. Myers, *Digital Signal Processing, Efficient Convolution and Fourier Transform Techniques*, Prentice-Hall, 1990.
6 L. R. Rabiner, R. W. Schafer, and C. M. Rader, The chirp z-transform algorithm, *IEEE Trans. Audio and Electroacoust.* AU-17(2):86 (June 1969).

7 K. R. Rao, P. Yip, *Discrete Cosine Transform: Algorithms, Advantages, Applications*, Academic, 1990.
8 R. Peters, F. Blischke, and H. Meyr, Transit-time estimation in turbulent flows, *IEEE Trans. Acoust. Speech Signal Process.* 36:1 (1988).

Received 15 April 1992; final manuscript accepted 7 February 1994